

# PROMPT : Placement Rapide Optimisé sur Machines Parallèles pour applications Télécoms

Michel Barreteau, Jean Jourdan, Juliette Mattioli<sup>1</sup>  
Corinne Ancourt, François Irigoien<sup>2</sup>  
Thierry Grandpierre, Christophe Lavarenne, Yves Sorel<sup>3</sup>  
Philippe Kajfasz<sup>4</sup>  
Bernard Dion<sup>5</sup>

## 1. Introduction

Les processeurs qui seront utilisés dans les années à venir sur les applications de type Large Bande, UMTS, ... exploiteront au mieux les possibilités offertes par les évolutions de la technologie silicium. Ils intégreront, sur une même puce, du traitement de signal systématique, du traitement de signal hétérogène, et du traitement de données.

Une première instantiation de ce type de System On a Chip (SOC) est MEFISTO [1], qui intègre des unités de calcul SIMD, une unité de traitement flottant (FPU), un processeur d'usage général ARM. Ce circuit, développé pour des applications telles que le traitement front-end pour les stations de base, possède des capacités de connexion en grappe lui permettant d'apporter une réponse matérielle simple pour supporter de très fortes puissances de calcul.

Actuellement, aucun environnement ne permet de développer de façon rapide et aisée les applicatifs s'exécutant sur des SOC's hétérogènes, tant au niveau du placement optimisé des données sur les structures SIMD, de la distribution et de l'ordonnancement optimisés des tâches que de la génération du code de communication inter-unités.

Cet article présente le projet PROMPT [2] dont l'objectif est de définir une démarche de conception générique, de type "Adéquation Algorithme Architecture" (AAA), pour les applications temps réel sur SOC, au travers d'un environnement de développement prototype qui intégrera l'ensemble des outils d'aide au placement et à la génération de code.

## 2. Aperçu de l'état de l'art

L'ensemble de la chaîne de conception matérielle de circuits de type SOC est actuellement couvert par des outils du commerce. En revanche, les environnements de développement associés pour programmer ces circuits sont inexistantes et seuls quelques outils, issus de la recherche, abordent certains points durs tels que le placement et l'optimisation des applications de traitement du signal (TS). Il existe différentes approches dont celles de Ptolemy et de Fx :

- L'environnement Ptolemy [3] permet la simulation, la modélisation, et la génération de code générique pour les applications de TS. Une fois le graphe de flot de données donné par l'utilisateur, différents algorithmes spécialisés selon la ressource à optimiser sont construits. Plusieurs machines cibles sont envisageables bien que les travaux soient principalement orientés vers un parallélisme de contrôle sur machines MIMD. Des algorithmes spécialisés sont généralement utilisés pour résoudre le problème : les méthodes d'ordonnements prédéfinies

---

<sup>1</sup> [Prénom.Nom@lcr.thomson-csf.com](mailto:Prénom.Nom@lcr.thomson-csf.com) , Thomson-CSF LCR, laboratoire A&TS, Domaine de Corbeville, F-91404 Orsay.

<sup>2</sup> [Prénom.Nom@cri.ensmp.fr](mailto:Prénom.Nom@cri.ensmp.fr) , Ecole des Mines de Paris CRI, F-77305 Fontainebleau.

<sup>3</sup> [Prénom.Nom@inria.fr](mailto:Prénom.Nom@inria.fr) , INRIA Rocquencourt, BP 105, F-78153 Le Chesnay.

<sup>4</sup> [Prénom.Nom@tcc.thomson-csf.com](mailto:Prénom.Nom@tcc.thomson-csf.com) , Thomson-CSF Communications, 66 rue du Fossé Blanc, BP 156, F-92231 Gennevilliers.

<sup>5</sup> [Prénom.Nom@simulog.fr](mailto:Prénom.Nom@simulog.fr) , SIMULOG, 1 rue James Joule, F-78286 Guyancourt.

permettent l'optimisation d'une ressource, comme la mémoire, la mémoire cache, le nombre de synchronisations, les communications ou l'optimisation de la latence. L'environnement Ptolemy ne traite donc pas le problème de l'optimisation globale du placement.

- Le compilateur Fx [4] adresse la même classe algorithmique de TS que celle définie précédemment. Le placement s'effectue à l'aide de primitives HPF. Ces travaux se placent dans des restrictions applicatives semblables à celles du traitement de signal systématique et propose une approche basée sur le parallélisme de données. Le compilateur Fx permet donc seulement d'estimer les performances d'un placement.

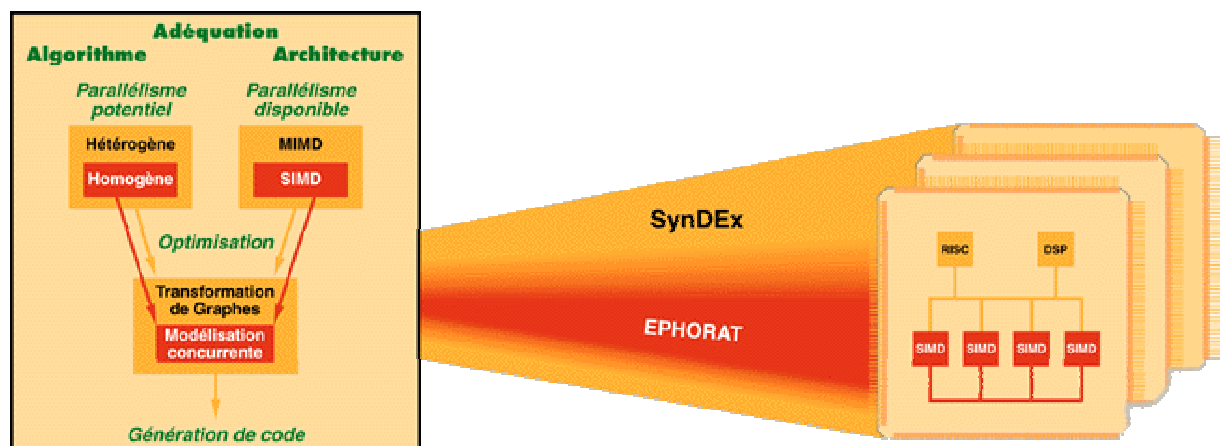
### 3. Technologies mises en œuvre

La méthodologie "Adéquation Algorithme Architecture" (AAA) et le logiciel SynDEx [5] qui la supporte sont parmi les plus avancés dans le domaine de la distribution et de l'ordonnancement des calculs et des communications sur architectures hétérogènes composées de différents types d'unités de calcul et de ressources de communication. Cette méthodologie est basée sur un modèle unifié de graphes factorisés, autant pour spécifier l'algorithme et l'architecture hétérogène que pour déduire les implantations optimisées en fonction des caractéristiques connues (mesurées) de l'architecture, et pour générer un macro-code d'exécutif aisément traduisible par un jeu de macros spécifiques à chaque type d'unité de calcul cible.

Pour les parties homogènes SIMD, la prise en compte, par la démarche PLC<sup>2</sup> [6] et le logiciel EPHORAT qui la supporte, de l'ensemble des ressources, de toutes les composantes du placement (partitionnement, alignement, communications, allocation, distribution des données) nécessite de combiner les différentes techniques de placement. Cette combinaison peut être réalisée par composition de contraintes, c'est-à-dire en introduisant les méthodes issues de la programmation par contraintes. Cette dernière offre un cadre cohérent, permettant la composition des contraintes via des connecteurs logiques ou règles de propagation.

### 4. Originalité du projet

L'originalité du projet consiste à prendre en compte simultanément les aspects homogènes SIMD et hétérogènes des applications télécoms, aussi bien au niveau des algorithmes et des architectures SOC et multi-SOC qu'au niveau des méthodes d'optimisation de l'implantation et de l'implantation elle-même (génération automatique de code compilable aisément portable).



Par rapport aux approches traditionnelles qui ne supportent le concepteur d'applications que pour la compilation de ses codes sources et au mieux pour une allocation dynamique des ressources (mémoire, temps CPU, routage des communications) et qui laissent à la charge du programmeur la distribution des calculs et son optimisation ainsi que le codage et le débogage de l'enchaînement des calculs et des communications, l'approche globale innovante du projet devrait permettre des gains de temps de développement importants, en éliminant la quasi-totalité des problèmes de mise au point de l'implantation distribuée des algorithmes (qui représentent habituellement une partie importante des

temps de développement), mais aussi en permettant une évaluation et une optimisation des performances des algorithmes et de leurs variantes avant même que l'architecture ne soit physiquement disponible, et en permettant un portage aisé des algorithmes sur de nouvelles variantes d'une architecture ou même sur d'autres architectures.

La recherche réalisée par la communauté de la parallélisation automatique de TS est principalement centrée sur la conception de nouveaux algorithmes et techniques pour la résolution séparée de chacune des fonctions nécessaires au placement d'applications sur machines parallèles. L'originalité de l'approche PLC<sup>2</sup> est d'essayer de traiter globalement et de façon concurrente les fonctions de partitionnement, d'alignement, de distribution et d'ordonnement en utilisant la démarche de modélisation concurrente que permet la programmation par contraintes. En effet, comme les langages de spécification d'applications de signal, les polyèdres en nombres entiers et les applications affines permettent de modéliser précisément les fonctionnalités du placement, l'algèbre linéaire permet de construire les différents modèles au niveau de granularité requis par la complexité du problème général, et les différents modèles du placement de traitement de signal systématique sur architecture homogène (type SIMD) peuvent s'exprimer à l'aide de contraintes linéaires et non linéaires intégrables au domaine de la programmation par contraintes.

L'originalité de l'étude de la partie homogène des SOC's est à la fois dans la démarche suivie et dans la prise en compte de l'impact de l'hétérogénéité des SOC's sur la partie homogène de l'architecture. Pour cela, il est nécessaire de modéliser les informations pertinentes sous forme de contraintes supplémentaires à imposer aux différents modèles du placement homogène.

## 5. Perspectives d'innovations

L'arrivée de nouveaux processeurs très intégrés de type SOC requiert une vision "système" des traitements or, les services niveau système ne sont disponibles actuellement que dans des outils de recherche.

La possibilité d'offrir, au terme du projet PROMPT, un environnement de développement intégré à même de prendre en compte la coopération entre les structures homogènes et hétérogènes permettra d'entreprendre le co-développement matériel-logiciel des nouvelles applications télécoms en apportant:

- ❖ une meilleure productivité:
  - aide au placement,
  - génération du code de communication inter-unités ;
- ❖ une adéquation fine entre l'application et son implantation matérielle, au travers d'une démarche coopérative intégrant :
  - placement optimisé des données pour les unités SIMD,
  - distribution et ordonnancement des tâches;
- ❖ ainsi qu'une qualité du logiciel améliorée :
  - génération des codes de communication inter-unité de calcul,
  - portage aisé sur de nouvelles cibles (macro-code)
- ❖ au travers d'un environnement générique permettant :
  - l'intégration de nouveaux modèles d'unités de calcul et de communication,
  - un format de représentation unique (macro-code).

## 6. Démonstrateur

Un démonstrateur, constitué de l'environnement prototype de développement réalisé dans le cadre du projet, validé par une application télécom typique implantée sur MEFISTO, premier SOC cible, permettra d'évaluer l'efficacité de la méthode employée aussi bien au niveau des performances des implantations obtenues qu'au niveau de la rapidité des cycles de développement.

Par souci de ne pas développer un outil de placement trop fortement dédié à une architecture cible particulière, il est important de rester le plus longtemps possible générique. C'est pourquoi, les sorties du placement se feront sous forme d'un macro-code qui devra s'interfacer avec les ateliers de développement (générateurs de code dédiés) spécifiques aux architectures cibles.

Ce projet devrait apporter des éléments pour permettre de définir une normalisation de ce macro-code, permettant une plus grande réutilisabilité de l'outil et autorisant ainsi un placement sur des différentes architectures SOC.

## 7. Conclusion

Cet article a présenté un projet dont l'objectif est de définir une démarche de conception générique, de type AAA, pour les applications temps réel sur SOC au travers d'un environnement de développement prototype qui intégrera l'ensemble des outils d'aide au placement et à la génération de code. L'originalité du projet consiste à prendre en compte simultanément les aspects homogènes SIMD et hétérogènes des applications télécoms, aussi bien au niveau des algorithmes et des architectures SOC et multi-SOC, qu'au niveau des méthodes d'optimisation de l'implantation et de l'implantation elle-même.

## Bibliographie

- [1] D. Maufroid, P. S. Paolucci, P. Kajfasz, A. Bertini. « mAgic FPU : VLIW Floating Point Engines for System-On-Chip Applications ». EMMSEC'99, Stockholm, Suède, Juin 1999.
- [2] [http://www.tcc.thomson-csf.com/our\\_act/tech/adv\\_archi/prompt\\_f.htm](http://www.tcc.thomson-csf.com/our_act/tech/adv_archi/prompt_f.htm)
- [3] <http://ptolemy.eecs.berkeley.edu>
- [4] <http://www.cs.cmu.edu/~fx/fx.html>
- [5] <http://www-rocq.inria.fr/syndex/pub.htm>
- [6] C. Ancourt, D. Barthou, C. Guettier, F. Irigoien, B. Jeannet, J. Jourdan, J. Mattioli. « Automatic Data Mapping of Signal Processing Applications ». IEEE International Conference on Application Specific Systems, Architectures, and Processors, Zurich, Suisse, pages 350-362, 14-16 Juillet 1997.